

Covering the Boundary of a Simple Polygon with Geodesic Unit Disks

George Rabanca and Ivo Vigan *

Department of Computer Science, City University of New York,
The Graduate Center, New York, NY 10016, USA.

Abstract. We consider the problem of covering the boundary of a simple polygon on n vertices using the minimum number of geodesic unit disks. We present an $O(n \log^2 n + k)$ time 2-approximation algorithm for finding the centers of the disks, with k denoting the number of centers found by the algorithm.

1 Introduction and Main Results

For two points u and v in a simple polygon P , the *geodesic distance*, denoted by $d(u, v)$, is the length of the shortest path between u and v inside P . A *geodesic unit disk* $D(v)$ centered at a point $v \in P$ is the set of all points in P whose geodesic distance to v is at most 1.

The *boundary* of $D(v)$, denoted by $\partial D(v)$, consists of all points in P which are either exactly at distance 1 from v or at distance at most 1 from v but contained in the polygon boundary ∂P . The *interior* of $D(v)$, denoted by $\text{int}(D(v))$, consists of all the points of $D(v)$ not contained on the boundary of $D(v)$, i.e., $\text{int}(D(v)) = D(v) \setminus \partial D(v)$, as shown in Fig. 1.

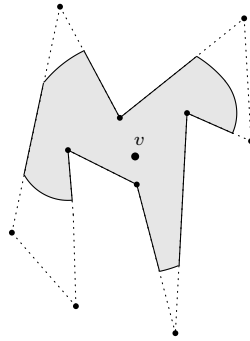


Fig. 1: A polygon (dotted) containing a geodesic disk centered at v , whose interior is depicted in gray and its boundary is drawn in black.

* Research supported by NSF grant 1017539

A collection of geodesic disks *covers* the polygon boundary ∂P , if each point of ∂P is contained in at least one disk. In this paper we present an $O(n \log^2 n + k)$ time 2-approximation algorithm which finds a collection of geodesic unit disks covering the boundary of a simple polygon on n vertices, with k denoting the number of disks found by the algorithm. The algorithm then returns the centers of the disks. We consider the setting where the centers can be placed anywhere inside the polygon, but the algorithm can be easily modified to restrict the centers to lie on ∂P . Furthermore, the *number* of disks can be computed in time $O(n \log^2 n)$.

While it follows from Theorem 7 of [29] that our problem is NP-hard in polygons with holes, its complexity remains open in simple polygons.

The main motivation for studying this problem comes from sensor networks, where *Barrier Coverage* problems have been studied extensively (see for example [5],[9],[10],[20],[21],[25],[26]). In a Barrier Coverage problem the goal is to place few sensors or guards to detect any intruder into a given region. The algorithm in this paper can be applied to this context: given a region, bounded by a piecewise linear closed border, such as a fence, place few guards inside the fenced region, such that wherever an intruder cuts through the fence, the closest guard is at most distance one away. Another way of looking at this problem is from an Art Gallery perspective (see for example [22]), where the polygon represents a gallery and, regardless where on the wall a painting is hanged, the closest guard is at most a distance one away.

1.1 Related Work

Several papers ([13],[17],[18],[19],[27],[30]) study full coverage of geometric regions with Euclidean disks. For an overview of optimal coverings of squares and triangles with disks see Chapter 1.7 of [7].

In the context of Barrier Coverage, [8] presents a polynomial time algorithm which for two points in the plane and a set of Euclidean disks selects a minimal subset of the disks which separates the two points. Extending the problem to k points, an $O(1)$ -approximation algorithm was presented in [14] and NP-hardness was shown in [23]. The same two point separation problem was studied in [1] when segments instead of disks are given.

Covering a simple polygon with a single geodesic disk of minimum radius has been studied in [24] and a linear-time algorithm is presented in [4]. An output sensitive algorithm for computing geodesic disks for a given set of centers and a fixed radius is presented in [6].

1.2 Paper Organization

This paper is organized as follows. In Section 2 we present the algorithm and show that it runs in time $O(n \log^2 n + k)$. In Section 3 we prove that the number of centers placed by the algorithm is at most twice the minimum number of centers needed to cover the polygon boundary. In Section 4 we show that a simple linear time algorithm achieves an asymptotically optimal approximation

ratio when the polygon perimeter is much larger than n . All the missing proofs can be found in the Appendix.

2 The Algorithm and Its Running Time

Our algorithm makes use of several properties of geodesic Voronoi diagrams which we review below.

2.1 Geodesic Voronoi diagrams

A *furthest-site geodesic Voronoi diagram* of k sites in a simple polygon P on n vertices is a decomposition of P into cells such that all points in a cell have the same site *furthest* away from them (in the geodesic metric). As shown in [3], it has combinatorial complexity $O(n + k)$ and can be constructed in time $O((n + k) \log(n + k))$. In Section 2.8 of [3] it is shown that these combinatorial and time complexities are with respect to a *refinement* (also called a shortest path partition) of the Voronoi edges. For all points on a refined edge it holds that their shortest paths to each of the two furthest sites are combinatorially equivalent, i.e., they consist of the same sequence of polygon vertices respectively. Furthermore, Section 3.3 of [3] defines for each of the $O(n + k)$ refined edges, and for each of the two furthest sites s_1 and s_2 defining a Voronoi edge e , the *anchor points* $a_e(s_1)$, $a_e(s_2)$ which are the last points on the shortest path from s_1 , s_2 respectively to any point on e . Those anchors can be computed in total $O(n + k)$ time and each time we compute a furthest-site geodesic Voronoi diagram we store the anchors as well as the distance to its site at the refined Voronoi edges. An additional property of this Voronoi diagram is that its edges form a tree, rooted at the *geodesic center* of the k sites, which is defined as the point that minimizes the maximum distance to any of the sites (see Corollary 2.9.3 of [3]). Therefore, the geodesic center of the sites can be obtained within the same time bound.

The second data structure we use is the *closest-site geodesic Voronoi diagram* which, for k sites in a simple polygon P on n vertices, is a decomposition of P into cells such that all points in a cell have the same site *closest* to them (in the geodesic metric). It has combinatorial complexity $O(n + k)$ and it can be constructed in time $O((n + k) \log(n + k))$ (see [2]).

2.2 The CONTIGUOUSGREEDY Algorithm

In this section we describe a greedy 2-approximation algorithm which finds a collection of geodesic unit disks which cover the boundary of P and returns the set of disk centers. It starts at vertex v_1 of the vertices v_1, \dots, v_n of P and iteratively extends a contiguous cover Γ of ∂P (in clockwise order) by the maximum amount that can be covered with a single geodesic disk. We denote the clockwise endpoint of Γ by c , thus initially $\Gamma = \{v_1\}$ and $c = v_1$.

We cover segment portions longer than 2 in time linear in the minimum number of disks needed to cover them. With v_u denoting the first uncovered vertex in the current iteration, we partially cover $\overline{cv_u}$ by adding $\lceil d(c, v_u)/2 \rceil - 1$ centers sequentially on $\overline{cv_u}$. By this, we assure that none of those disks contains v_u and, since each disk contains a boundary portion of length 2, the disks placed are indeed optimal with respect to the greedy contiguous extension criterion.

Definition 1. For a polygonal chain C , we denote by $\|C\|$ the sum of the lengths of its line segments and we refer to the number of vertices of C by $|C|$.

Definition 2. For two points $u, v \in \partial P$, we denote the portion of ∂P in clockwise orientation between u and v by $\partial P[u, v]$.

If c does not lie on a long segment, we compute the next endpoint c' which extends Γ in clockwise order by a maximum length boundary portion which can be covered by a single geodesic unit disk. We do this by finding the first vertex v_u (in clockwise order) such that $\partial P[c, v_u]$ cannot be contained in a single geodesic unit disk. This test is done by calling the `TESTCOVER`(c, v) procedure discussed below, which, for a boundary point c and a vertex v tests whether $\partial P[c, v]$ can be covered with a single geodesic unit disk. If v_i is the first vertex in clockwise order after c , we find v_u by first using exponential search with the `TESTCOVER` predicate with c fixed and v set to $v_{i+1}, v_{i+2}, v_{i+4}, \dots, v_{i+2^k}, \dots$ respectively in consecutive steps until `TESTCOVER` returns *false* or $i + 2^k > n$. This defines an index-interval containing the index u which can then be found using a simple binary search.

After finding v_u and thereby fully determining the sequence of vertices covered in the current iteration, we use the `AUGMENTSHORT` procedure – discussed below – to compute the new endpoint c' of Γ as well as the center of the next disk.

CONTIGUOUSGREEDY

$c \leftarrow v_1, v_u \leftarrow v_2$

$S \leftarrow \emptyset$

while ∂P not covered:

1. If $\overline{cv_u}$ is longer than 2

compute centers on $\overline{cv_u}$ at steps of 2; add them to S ; update c

2. Update v_u to the first vertex s.t. $\partial P[c, v_u]$ cannot be covered by a single disk, using Exponential and Binary Search with predicate `TESTCOVER`

3. Use `AUGMENTSHORT` to cover the vertices between c and v_u , and a maximal portion of the edge $\overline{v_{u-1}v_u}$; add new center to S and update c

end while

return S

Definition 3. For two points u, v in a simple polygon P , we denote the shortest path in P between u and v by $\pi(u, v)$. We denote the number of its vertices by $|\pi(u, v)|$.

Definition 4 ([28]). A set Q inside a simple polygon P is called *geodesic convex*, if for any two points $u, v \in Q$, the shortest path $\pi(u, v)$ is contained in Q .

TESTCOVER(c, v). This procedure tests for a boundary point c and a polygon vertex v whether $\partial P[c, v]$ can be covered with a single geodesic unit disk. Observe that if a geodesic unit disk can cover a set of points, then a geodesic unit disk centered at the *geodesic center* of those points obviously also covers them. Let $U = U(c, v)$ denote the sequence of point c and all polygon vertices up to (and including) v in clockwise order. TESTCOVER computes the geodesic center of U and returns true iff it has distance at most one to all points in U .

Implementation details. We compute the geodesic center of U in a smaller polygon Q containing U . We let Q be $\partial P[c, v] \circ \pi(v, c)$, with \circ denoting the concatenation of two polygonal chains sharing two endpoints. Note that Q may have touching sides, but it is not self-intersecting. Such polygons are referred to as *weakly simple* polygons ([12]) and the geodesic distance within them is well defined. Since Q is the concatenation of a boundary part of P and a shortest path in P it follows that Q is geodesic convex in P , thus implying that the geodesic center of U in Q is the same point as the geodesic center of U in P . We find this geodesic center point by computing the furthest-site geodesic Voronoi diagram $\mathcal{VP}_Q(U)$ of the sites U in Q , traversing the (oriented) Voronoi edges to the root and thereby obtain the geodesic center of U (see Section 2.1). Then, for each site in U we test whether the distance to the geodesic center is at most one.

Computational complexity. Computing $\pi(v, c)$ takes time $O(|\pi(v, c)| \log n)$ after $O(n)$ global pre-processing time, using the algorithm of [16]; concatenating two polygonal chains to construct Q takes constant time. Computing $\mathcal{VP}_Q(U)$ takes $O(|Q| \log(|Q|))$ time and the geodesic center can be obtained from $\mathcal{VP}_Q(U)$ in the same time bound. Computing the distance from the geodesic center to all sites in U can be done in time $O(|Q|)$ (see [15]), by building the shortest path tree rooted at the geodesic center. Therefore, the procedure has an overall time complexity of $O(|Q| \log n)$.

Knowing the first vertex v_u such that $\partial P[c, v_u]$ cannot be covered with a single geodesic unit disk, we compute the center of the next disk and compute the new endpoint c' of Γ the following AUGMENTSHORT procedure.

AUGMENTSHORT(c, v_u). For the new endpoint c' of Γ it needs to hold that $\partial P[c, c']$ can be covered with one geodesic unit disk, and for any $c'' \in \partial P$, with $\|\partial P[c, c']\| > \|\partial P[c, c'']\|$ it is not possible to cover $\partial P[c, c'']$ with a single geodesic unit disk. Let $\overline{U} = \overline{U}(c, v_{u-1})$ denote the clockwise sequence of point c and all vertices up to (and including) v_{u-1} . We construct $Q = \partial P[c, v_u] \circ \pi(v_u, c)$ and denote by A the intersection of the geodesic unit disks centered at the points \overline{U} in Q . This intersection is non-empty by construction and the center of the next disk lies in A . We denote by I the set of all disk-disk intersection points on ∂A as shown in Fig. 2. Lemma 1 below justifies the steps taken to find c' .

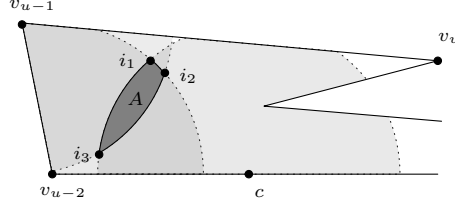


Fig. 2: Illustration of A , i.e., the intersection of the geodesic unit disks centered at points in $\overline{U} = \{c, v_{u-2}, v_{u-1}\}$ as well as the disk-disk intersection points $I = \{i_1, i_2, i_3\}$.

Lemma 1. *Given a simple polygon P , let A be the non-empty intersection of a collection of geodesic unit disks in P and let $\overline{\alpha\beta}$ be a line-segment in P , such that for all $a \in A$, $d(a, \alpha) \leq 1$ and $d(a, \beta) > 1$. For any point $c \in \overline{\alpha\beta}$ and any disk center q furthest away from c , $d(c, A) = 1$ if and only if either:*

- a) $d(c, q) = 2$ and $\pi(c, q) \cap \partial D(q) \in A$, or
- b) $d(c, I) = 1$ and $\pi(c, q) \cap \partial D(q) \notin A$,

with I denoting the disk-disk intersection points on ∂A and $d(c, Y) = \min_{s \in Y} d(c, s)$, for a point set Y in P .

Proof. (Lemma 1) Let p be the point in A closest to the point c and by q a center farthest from c . Notice that since A is geodesic convex, p is unique and it lies on ∂A . We prove Lemma 1 with the help of the following two observations.

Observation 1 *If $p \in \partial D(q')$ for some center q' and $\pi(q', c) \cap \partial D(q') \notin A$ then $p \in I$.*

Proof. Let $p' = \pi(q', c) \cap \partial D(q')$ and assume that $p \notin I$, thus p is in the interior of all the disks defining A , other than $D(q')$. Furthermore, since $D(q')$ is geodesic convex, $\pi(p, p') \cap A$ contains a point a , with $a \neq p$. Since $d(p', q') = d(p, q') = 1$ and p' is on the shortest path from q' to c , by uniqueness of the shortest path, $d(c, p') < d(c, p)$. Then, by Lemma 2, $d(c, a) < \max\{d(c, p), d(c, p')\} = d(c, p)$, contradicting that p is the point in A closest to c . \square

Observation 2 *If $p \in A \setminus I$ then $p = \pi(c, q) \cap \partial D(q)$.*

Proof. Since $p \notin I$ there is a unique center q' such that $p \in \partial D(q')$. As shown in Observation 1, if $p \in D(q')$ and $\pi(c, q') \cap D(q') \notin A$ then $p \in I$. Therefore $\pi(c, q') \cap D(q')$ is contained in A and we denote this point by p' .

Observe that p' is contained in $\pi(c, q')$ and is at distance 1 from q' , thus $d(q', c) = d(q', p') + d(p', c) = 1 + d(p', c)$. Clearly, $d(q', c) \leq d(q', p) + d(p, c) = 1 + d(p, c)$ and since p is the closest point in A to c , it follows that $p = p'$. Now observe that, since $d(q', c) = 1 + d(p, c)$ and the distance between p and any other center is less than 1 (because $p \notin I$), q' is the farthest center, i.e., $q' = q$. Therefore $p = \pi(c, q) \cap \partial D(q)$ holds as claimed. \square

We now prove Lemma 1 :

" \Rightarrow ". We distinguish two cases based on whether $p \in A \setminus I$ or $p \in I$. If $p \in A \setminus I$, then $p = \pi(c, q) \cap D(q)$ as shown in Observation 2. Therefore, $\pi(c, q) \cap D(q) \in A$ and $d(c, q) = d(c, p) + d(p, q) = 2$, and thus condition a) holds. If $p \in I$ and $\pi(c, q) \cap D(q) \notin A$ condition b) holds. Otherwise if $p \in I$ and $\pi(c, q) \cap D(q) \in A$, let $p' = \pi(c, q) \cap D(q)$. We can write the distance $d(c, q)$ as $d(c, p') + d(p', q)$. Since p' lies on $\partial D(q)$, $d(p', q) = 1$ and since $d(c, A) = 1$ this implies that $d(c, p') \geq 1$. Therefore, distance $d(c, q) \geq 2$.

By the triangle inequality it also holds that $d(c, q) \leq d(c, p) + d(p, q)$. Since p is the closest point in A to c , $d(c, p) = 1$ by hypothesis. Since p lies on $\partial D(q)$, $d(p, q) = 1$. Therefore, $d(c, q) \leq 2$ and combining this with $d(c, q) \geq 2$ from above, $d(c, q) = 2$ and again condition a) holds.

" \Leftarrow ". a) Let $p' = \pi(c, q) \cap \partial D(q)$. Then $d(c, A) \leq d(c, p') = d(c, q) - d(p', q) = 1$. If $d(c, A) < 1$, by definition $d(c, p) < 1$. Since $p \in A$, $d(p, q) \leq 1$ and by the triangle inequality, $d(c, q) \leq d(c, p) + d(p, q) < 2$ which contradicts $d(c, q) = 2$. b) Since $d(c, I) \leq 1$ and $I \subseteq A$, obviously $d(c, A) \leq 1$. For $p \in A$ the closest point to c in A , assume that $d(c, p) < 1$. Since $d(c, I) = 1$, $p \notin I$. Therefore, by Observation 2, $p = \pi(c, q') \cap \partial D(q')$, and thus this intersection is in A contradicting the hypothesis. \square

We use the following steps to determine c' on $e = \overline{v_{u-1}v_u}$.

- Step 1) Find the point x_1 on e closest to v_u , whose distance to its furthest point q in \overline{U} is exactly 2 and $\pi(x_1, q) \cap \partial D(q) \in A$, if such a point x_1 exists.
- Step 2) Find the point x_2 on e closest to v_u , whose distance to its closest point in I is exactly 1.
- Step 3) Set $c' \leftarrow x_2$ if x_1 does not exist or $d(x_2, v_u) < d(x_1, v_u)$. In this case we add the point in I closest to c' as the new disk center to the set S of centers. Otherwise $c' \leftarrow x_1$ and the point $\pi(x_1, q) \cap \partial D(q)$ is the new disk center which gets added to S .

Note that since v_{u-1} will be covered in this iteration and v_u won't be covered, $d(v_{u-1}, A) \leq 1 < d(v_u, A)$. By continuity of the geodesic distance, there is a point c' on e , with $d(c', A) = 1$ and thus by Lemma 1 either x_1 or x_2 exists.

In Step 1, to find x_1 if it exists, we construct the (refined) furthest-site geodesic Voronoi diagram of the sites \overline{U} in Q and traverse the Voronoi vertices $\gamma_1, \dots, \gamma_m$ on e , ordered in the direction from v_u to v_{u-1} and set $\gamma_{m+1} = v_{u-1}$. For each such vertex we check in $O(\log |Q|)$ time whether the distance to (one of) its furthest site(s) is at most 2, using an $O(\log |Q|)$ time shortest path query ([16]) after pre-processing Q in $O(|Q|)$ time. Once we find the first γ_j with distance at most 2, if it exists, this determines a sub-segment $\overline{\gamma_j \gamma_{j-1}}$ on e containing a point x at distance exactly 2 from its furthest site q . Note that since the shortest paths to the furthest site q have the same combinatorial structure for all points on the refined Voronoi edge $\overline{\gamma_j \gamma_{j-1}}$, we find the point at distance 2 to q in constant time since we stored the anchor point $a_{\overline{\gamma_j \gamma_{j-1}}}(q)$ at the edge $\overline{\gamma_j \gamma_{j-1}}$ (see Section 2.1). We check if $\pi(x, q) \cap \partial D(q) \in A$, by computing $D(q)$ in time

$O(|Q|)$ using [15] and finding in $O(\log |Q|)$ time the arc α of $D(q)$ separating q from x . We traverse the edges of $\pi(x, q)$ and for each edge we test in $O(1)$ time if it intersects α . Denoting the intersection point by p , we check if $p \in A$, by computing the shortest path tree to the sites in U and test if the distance to all sites is at most 1 in time $O(|Q|)$. If this intersection is in A , we set x_1 to x .

Claim. There can be at most two points on e that have distance exactly 2 from their respective furthest site; if there are two such points, one of them must be v_{u-1} .

We prove this claim using the following lemma.

Lemma 2 (Lemma 1 [24]; see also Lemma 2.2.1 [3]). *Given three points a, b, c in a simple polygon, for $x \in \pi(b, c)$, the distance $d(a, x)$ is a convex function on $\pi(b, c)$, with $d(a, x) < \max\{d(a, b), d(a, c)\}$.*

Proof (of Claim). Assume that there are two points p_1 and p_2 on $e \setminus \{v_{u-1}\}$ that are at distance 2 from their respective furthest sites, with p_1 closer to v_{u-1} than p_2 , thus $p_1 \in \overline{v_{u-1}p_2} \setminus \{v_{u-1}, p_2\}$. Let q_1 be a center furthest away from p_1 . Clearly $d(q_1, v_{u-1}) \leq 2$ since both q_1 and v_{u-1} are at distance at most 1 from any point in A . Since $d(q_1, p_2) \leq 2$ and $p_1 \in \overline{v_{u-1}p_2} \setminus \{v_{u-1}, p_2\}$, by Lemma 2 $d(q_1, p_1) < 2$, contradicting the assumption that $d(q_1, p_1) = 2$. \square

According to the above claim, the only other candidate for x_1 is v_{u-1} . Thus, if $\pi(x_1, q) \cap \partial D(q) \notin A$ we check in $O(\log |Q|)$ time if the point v_{u-1} is at distance exactly 2 from its furthest site and if so, we set x_1 to v_{u-1} . If x_1 exists $\partial P[c, x_1]$ can be covered with one geodesic unit disk, because the point $\pi(x_1, q) \cap \partial D(q)$ has distance exactly 1 to x_1 and lies in A .

In Step 2, to find x_2 , we first construct the set I of the disk-disk intersection points of A ; we do this without explicitly computing A . To construct I , we look at the furthest-site geodesic Voronoi diagram of the sites \overline{U} in Q constructed in the Step 1. Since any point in I has two points in \overline{U} at distance 1, every point in I lies on a Voronoi edge. For every site $s \in \overline{U}$ we look at the refined edges of $\sigma(s)$ and for such edge e we access its anchor point $a_e(s)$ as well as the distance from s to the endpoints of e , in constant time. We test if there is a point on e having distance 1 to s , again in $O(1)$ time. If such a point exists then this is a disk-disk intersection point and we add it to I . Since we need constant time for each refined Voronoi edge, I can be computed in total time $O(|Q|)$.

Having computed I , we construct the closest-site geodesic Voronoi diagram of the sites I in Q . We traverse the Voronoi vertices $\gamma_1, \dots, \gamma_m$ on e , ordered in the direction from v_u to v_{u-1} and set $\gamma_{m+1} = v_{u-1}$. For each such vertex we check whether the distance to (one of) its closest site(s) is at most 1 again by an $O(\log n)$ time shortest path distance query. Once we find the first such vertex γ_j on $e = \overline{v_{u-1}v_u}$, if it exists, we have determined a sub-segment $\overline{\gamma_j \gamma_{j-1}}$ on e where x_2 lies. Letting $i \in I$ be the corresponding closest site, by Lemma 2, we find the point in $\overline{\gamma_j \gamma_{j-1}}$ at distance 1 from i by computing the intersection point of

a geodesic unit disk centered at i with $\overline{\gamma_j \gamma_{j-1}}$, in time $O(|Q|)$, using the funnel algorithm of [15].

There can be at most two points on e that have distance exactly 1 from i ; if there are two such points, one of them must be v_{u-1} . This can be seen directly from the fact that $d(i, v_{u-1}) \leq 1$, and Lemma 2. We set x_2 to the one closer to v_u . It is easy to see that x_2 is feasible, i.e., $\partial P[c, x_2]$ can be covered with one geodesic unit disk, because $d(i, \overline{U}) \leq 1$ and $d(i, x_2) = 1$.

In Step 3, $c' \leftarrow x_2$, if either x_1 does not exist or $d(x_2, v_u) < d(x_1, v_u)$, thus c' indeed extends Γ maximally because x_2 is the point on e closest to v_u having distance exactly 1 to the closest point in I , i.e., to the center of the geodesic unit disk placed in this iteration. Otherwise $c' \leftarrow x_1$ and x_1 is the point on e closest to v_u having distance exactly 2 to the furthest center in \overline{U} ; any point on e closer to v_u has distance larger than 2 from that center and is thus infeasible.

Computational Complexity / Summary. Constructing Q takes time $O(|Q| \log n)$ as argued in the `TESTCOVER`(c, v) paragraph before. Step 1 needs $O(|Q| \log |Q|)$ time to construct the geodesic furthest-site Voronoi diagram of \overline{U} in Q and $O(|Q| \log |Q|)$ time to find a sub-segment of the edge e possibly containing x_1 , since there are only $O(|Q|)$ Voronoi vertices in total and we spend $O(\log |Q|)$ on them for finding the sub-segment. The last step is to test if $\pi(x, q) \cap \partial D(q) \in A$, which takes time $O(|Q|)$ as argued above.

In Step 2, we spend $O(|Q|)$ time to construct the set I and $O(|Q| \log |Q|)$ time to construct the geodesic closest-site Voronoi diagram of the sites I . We then traverse edge e in $O(|Q| \log |Q|)$ time to find a sub-segment of the edge e possibly containing x_2 , and determine x_2 on this sub-segment in $O(|Q|)$ time.

Thus the overall time spent in `AUGMENTSHORT` is $O(|Q| \log n)$.

Total Running Time. Let \mathcal{Q} be the set of all polygons constructed throughout the whole execution of `CONTIGUOUSGREEDY`. In each polygon $Q \in \mathcal{Q}$ we spend $O(|Q| \log n)$ time in `TESTCOVER` and possibly $O(|Q| \log n)$ time in `AUGMENTSHORT` as argued above. Since in each iteration of `CONTIGUOUSGREEDY`, Γ is extended to cover at least one new polygon vertex, there are at most n iterations of the main *while* loop. Furthermore, covering long segments of ∂P takes total time $O(k)$. Since according to Lemma 3, $\sum_{Q \in \mathcal{Q}} |Q| = O(n \log n)$, the running time of `CONTIGUOUSGREEDY` is $O(n \log^2 n + k)$.

Lemma 3. $\sum_{Q \in \mathcal{Q}} |Q| = O(n \log n)$.

Proof. Each polygon of \mathcal{Q} constructed in the `CONTIGUOUSCOVER` algorithm has the form $Q = \partial P[c, w] \circ \pi(w, c)$, with c an arbitrary point on ∂P and w a vertex of P . We call $\partial P[c, w]$ the ∂ -portion and $\pi(c, w)$ the π -portion of the polygon Q . Notice that every polygon constructed in `AUGMENTSHORT` was also constructed in a `TESTCOVER` call and thus it suffices to bound the number of polygons constructed in all `TESTCOVER` calls.

Observe that $|\mathcal{Q}| = O(n \log n)$, since in each iteration, Γ is extended to cover at least one new vertex, thus there are at most n iteration, and in each iteration we construct $O(\log n)$ polygons during Exponential and Binary Search. Observe that if every vertex of P is contained in $O(\log n)$ polygons of \mathcal{Q} then $\sum_{Q \in \mathcal{Q}} |Q| = O(n \log n)$. This holds because for each $Q \in \mathcal{Q}$ there is at most one vertex of Q which is not a vertex in P , namely the point c .

Since v_1 is covered both in the first and last iteration of the algorithm, we are pessimistically bounding the number of polygons containing v_1 by $|\mathcal{Q}| = O(n \log n)$. To then prove the lemma it is enough to show that every vertex of P except v_1 is contained in $O(\log n)$ polygons of \mathcal{Q} . For that we fix a vertex v_k , with $1 < k \leq n$, and show that v_k appears in the ∂ -portion of $O(\log n)$ polygons and v_k appears in the π -portion of $O(\log n)$ polygons of \mathcal{Q} .

To bound the number of appearances of v_k on the ∂ -portion of a polygon we fix the unique iteration i^* in which v_k is first covered. Since TESTCOVER is used as a predicate in Exponential and Binary search, in iteration i^* it is called $O(\log n)$ times and thus v_k appears in $O(\log n)$ polygons during this iteration. Observe, that in subsequent iterations, when $i > i^*$, vertex v_k is not part of the ∂ -portion of any constructed polygon. For an iteration $i < i^*$, let v_{u_i} be the first uncoverable vertex (denoted by v_u in the algorithm) found in iteration i , thus $u_i \leq k$; let q_i be the number of polygons in which v_k appears on the ∂ -portion during this iteration i . Also observe that u_{i-1} is the index of the first vertex of P covered in iteration i . We claim that

$$k - u_i \leq \frac{k - u_{i-1}}{2^{q_i-1}}, \text{ for any } 1 \leq i < i^* \quad (1)$$

and defining $u_0 = 1$, implies that $\sum_{i=1}^{i^*-1} q_i \leq \log k \leq \log n$.

For $q_i = 0$, inequality (1) holds trivially. Otherwise, since v_k is not covered during this iteration, Exponential Search stops after the first time v_k appears on the ∂ -portion of a constructed polygon. This leaves a search interval of size at most $k - u_{i-1}$. During Binary Search, there are exactly $q_i - 1$ search intervals which contain both v_u and v_k . Since the interval size is halved at each step and all search intervals containing both v_{u_i} and v_k have size at least $k - u_i$, inequality (1) follows.

So far we have shown that v_k appears on the ∂ -portion of $O(\log k)$ polygons in \mathcal{Q} before iteration i^* , $O(\log n)$ times during iteration i^* and does not appear in subsequent iterations. Therefore, all together, v_k appears on ∂ -portions of $O(\log n)$ polygons in \mathcal{Q} .

To bound the number of appearances of v_k on the π -portion of a polygon, let $\mathcal{Q}_k \subseteq \mathcal{Q}$ be the set of polygons containing v_k on their π -portion but not on the ∂ -portion. By Observation 3 below, any two polygons in \mathcal{Q}_k intersect on their ∂ -portion because they both contain v_k on their π -portion. Since by construction the ∂ -portion of each polygon Q ends with a vertex, any two polygons in \mathcal{Q}_k have a vertex in common on their ∂ -portion. This is true because the ∂ -portion of those polygons are subsequences of (v_1, \dots, v_n, v_1) and it is easy to see that there is a vertex $v_{k'}$ that belongs to the ∂ -portion of all $Q \in \mathcal{Q}_k$. Since $v_{k'}$ appears on ∂ -portions of $O(\log n)$ polygons, $|\mathcal{Q}_k| = O(\log n)$. \square

Observation 3 For a, b, c, d four distinct points on ∂P , if $\pi(a, b) \cap \pi(c, d)$ contains a polygon vertex not contained in $\partial P[a, b] \cup \partial P[c, d]$, then $\partial P[a, b] \cap \partial P[c, d] \neq \emptyset$.

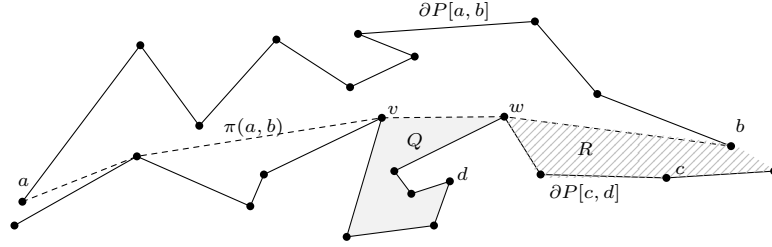


Fig. 3: Illustration of the proof of Observation 3.

Proof. Let v be a vertex contained in $\pi(a, b) \cap \pi(c, d)$ not in $\partial P[a, b] \cup \partial P[c, d]$ and assume for contradiction that $\partial P[a, b]$ and $\partial P[c, d]$ are disjoint. Then either $\partial P[c, d] \subset \partial P[a, v]$ or $\partial P[c, d] \subset \partial P[v, b]$. W.l.o.g. assume $\partial P[c, d] \subset \partial P[v, b]$. For w the successor vertex of v in $\pi(a, b)$, let Q be the simple polygon bounded by $\overline{vw} \circ \partial P[w, v]$. If both c, d are contained in Q , meaning $w \notin \partial P[c, d]$, since v is a convex vertex in Q , it holds that $v \notin \pi(c, d)$, a contradiction. Otherwise, let R be the geodesic convex set bounded by $\pi(w, b) \circ \partial P[b, w]$. If both c, d are contained in R , then by geodesic convexity, $\pi(c, d) \subseteq R$ and thus $v \notin \pi(c, d)$. Otherwise $d \in Q$ and $c \in R$ as shown in Fig. 3. Since in that case $Q \cap R = \{w\}$ and $R \cup Q$ is again a geodesic convex set, $\pi(c, d) = \pi(c, w) \circ \pi(w, d)$. Again, since v is a convex vertex in Q , $v \notin \pi(w, d)$, and thus v not in $\pi(c, d)$, a contradiction. \square

3 Approximation Ratio

Let OPT denote a set of geodesic unit disks optimally covering ∂P . In order to prove the 2-approximation we prove the existence of a *coloring* for ∂P using $|OPT|$ distinct colors and introducing at most $\max\{2|OPT| - 2, 1\}$ monochromatic boundary portions. We then show that CONTIGUOUSGREEDY uses at most one disk per monochromatic boundary portion (plus possibly one additional disk for the unique monochromatic boundary portion containing v_1), which implies the 2-approximation factor of CONTIGUOUSGREEDY.

A *coloring* of ∂P is a function $\gamma : \partial P \rightarrow \mathbb{N}$. The number of colors used by γ is defined as the cardinality of the image of γ . A *block* is a connected component of ∂P colored with a single color. We let ∂P_i denote the subset of the polygon boundary colored with color i and we call each connected component of $\partial P \setminus \partial P_i$ a *pocket* of ∂P induced by color i (see Fig. 4(b)).

A coloring of ∂P is called *crossing-free* if for any two distinct colors i, j , it holds that ∂P_j is contained in a single pocket induced by color i .

For a collection $\mathcal{D} = \{D_1, \dots, D_k\}$ of disks covering ∂P , a *disk-coloring* of ∂P w.r.t. \mathcal{D} is a function $\gamma_{\mathcal{D}} : \partial P \rightarrow \{1, \dots, k\}$, such that $\gamma(x) = i \Rightarrow x \in D_i$, i.e., a point on ∂P can only be colored with one of the indices of the disks covering it (see Fig. 4(a)).

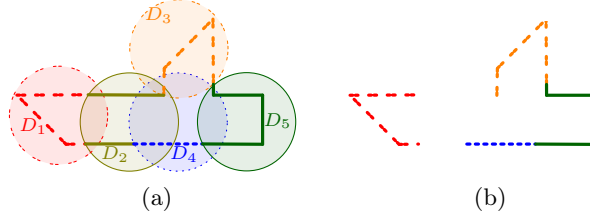


Fig. 4: (a) A crossing disk-free coloring of ∂P . (b) The two pockets induced by color 2.

Definition 5. For a coloring γ , two of its colors r and b cross each other, if there are two pockets induced by color r containing blocks of color b .

Observe that if two colors r and b cross each other, there are at least two blocks B_r^1, B_r^2 of color r and two blocks B_b^1, B_b^2 of color b such that sequence of blocks $B_r^1, B_b^1, B_r^2, B_b^2$ occurs in clockwise order on ∂P as shown in Fig. 5.

Lemma 4. In any disk-coloring, if two colors r and b cross each other, one of the following holds: 1) There exists a pocket induced by color r which contains blocks of color b and all these blocks can be re-colored with color r , s.t. the resulting coloring is still a disk-coloring. 2) There exists a pocket induced by color b which contains blocks of color r and all these blocks can be re-colored with color b , s.t. the resulting coloring is still a disk-coloring.

Proof. Suppose this is not possible. Since neither B_r^1 nor B_r^2 can be colored with b , there are points $\alpha_r \in B_r^1$ and $\beta_r \in B_r^2$ which lie outside of disk D_b . If we denote the center of D_b by c_b , it therefore holds that $d(c_b, \alpha_r) > 1$ and $d(c_b, \beta_r) > 1$ (see Fig. 5). Analogously, there are two points $\alpha_b \in B_b^1$ and $\beta_b \in B_b^2$, s.t. α_b and β_b can not be colored with color r . This again implies that both points lie outside of disk D_r centered at c_r and thus $d(c_r, \alpha_b) > 1$ and $d(c_r, \beta_b) > 1$.

Lemma 5. For any collection of disks covering ∂P , there exists a crossing free disk-coloring of ∂P .

Proof. Consider the four paths $\pi(c_r, \alpha_r)$, $\pi(c_r, \beta_r)$, $\pi(c_b, \alpha_b)$ and $\pi(c_b, \beta_b)$. Due to the alternating arrangement of the four blocks $B_r^1, B_b^1, B_r^2, B_b^2$ – and therefore of $\alpha_r, \alpha_b, \beta_r, \beta_b$, on the polygon boundary, one of the paths from c_r must intersect with one of the paths from c_b . Assume w.l.o.g. that $\pi(c_r, \alpha_r)$ intersects $\pi(c_b, \beta_b)$ and let p be an intersection point. Again, w.l.o.g., assume that $d(c_r, p) \leq d(c_b, p)$. Then, by the triangle inequality $d(c_r, \beta_b) \leq d(c_b, \beta_b) \leq 1$ contradicting our assumption that $d(c_r, \beta_b) > 1$. \square

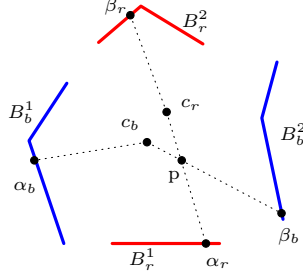


Fig. 5: Illustration of the four alternating blocks $B_r^1, B_b^1, B_r^2, B_b^2$ and the corresponding points α_r, β_r and α_b, β_b ; the disk centers c_r and c_b , as well as the intersection point p of $\pi(c_r, \alpha_r)$ and $\pi(c_b, \beta_b)$.

We are now going to prove Lemma 5, which states that for any collection of disks covering ∂P , there exists a crossing free disk-coloring of ∂P .

For a given disk-coloring w.r.t. a collection of disks \mathcal{D} , we let l_{ij} be the number of pockets induced by color i which contain blocks of color j ; observe that $l_{ij} = l_{ji}$. We refer to

$$\sum_{1 \leq i < j \leq |\mathcal{D}|} (l_{ij} - 1)$$

as the *crossing number* of the disk-coloring. By definition it holds that the crossing number of a coloring is zero if and only if the coloring is crossing free. We now let $\gamma = \gamma_{\mathcal{D}}$ be a disk-coloring of ∂P w.r.t. disks \mathcal{D} , having minimum crossing number (over all disk-colorings w.r.t. \mathcal{D}). Assume for contradiction that the crossing number of γ is not zero and let r and b be two colors of γ which cross each other, i.e., $l_{rb} = l_{br} \geq 2$. Then, w.l.o.g., according to Lemma 4, there exists a pocket \mathcal{P}_b induced by color b , in which all blocks of color r can be colored with b and the coloring remains a valid disk-coloring w.r.t. \mathcal{D} . We refer to the resulting disk-coloring as $\hat{\gamma}$ and by \hat{l}_{ij} to the number of pockets induced by color i of $\hat{\gamma}$ which contain blocks of color j (again in $\hat{\gamma}$). Lastly we denote by $\hat{\mathcal{P}}_r$ the pocket induced by color r in $\hat{\gamma}$, fully containing \mathcal{P}_b as shown in Figure 6.

We are going to show that $\hat{\gamma}$ has a smaller crossing number than γ , thus contradicting the assumption that γ is the disk-coloring with minimum crossing number. For this, we extend the definition of l_{ij} to parts of the polygon boundary: for a contiguous subset ∂Q of ∂P , we denote by $l_{ij}[\partial Q]$ the number of pockets induced by color i which are fully contained in ∂Q and which contain blocks of color j .

For the rest of the proof, let k be an arbitrary color of γ (and thus also of $\hat{\gamma}$). Since every pocket induced by color r in γ (and in $\hat{\gamma}$) is either contained in \mathcal{P}_r or in $\partial P \setminus \mathcal{P}_r$, it holds that

$$l_{rk} = l_{rk}[\hat{\mathcal{P}}_r] + l_{rk}[\partial P \setminus \hat{\mathcal{P}}_r] \quad \text{and} \quad \hat{l}_{rk} = \hat{l}_{rk}[\hat{\mathcal{P}}_r] + \hat{l}_{rk}[\partial P \setminus \hat{\mathcal{P}}_r]. \quad (2)$$

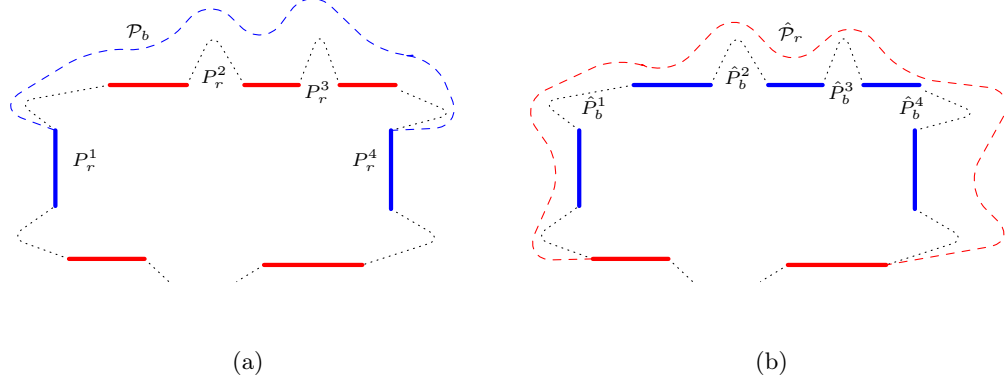


Fig. 6: (a) Illustration of \mathcal{P}_b in the disk-coloring γ . (b) Illustration of $\hat{\mathcal{P}}_r$ in the disk-coloring $\hat{\gamma}$.

Similarly, it holds that

$$l_{bk} = l_{bk}[\mathcal{P}_b] + l_{bk}[\partial P \setminus \mathcal{P}_b] \quad \text{and} \quad \hat{l}_{bk} = \hat{l}_{bk}[\mathcal{P}_b] + \hat{l}_{bk}[\partial P \setminus \mathcal{P}_b]. \quad (3)$$

Furthermore, since $\hat{\gamma}$ does not differ from γ in $\partial P \setminus \mathcal{P}_b$, it holds that

$$\hat{l}_{bk}[\partial P \setminus \mathcal{P}_b] = l_{bk}[\partial P \setminus \mathcal{P}_b] \quad (4)$$

and analogously, since $(\partial P \setminus \hat{\mathcal{P}}_r) \subseteq (\partial P \setminus \mathcal{P}_b)$, it holds that

$$\hat{l}_{rk}[\partial P \setminus \hat{\mathcal{P}}_r] = l_{rk}[\partial P \setminus \hat{\mathcal{P}}_r]. \quad (5)$$

Next, we are going to show that

$$\hat{l}_{rk}[\hat{\mathcal{P}}_r] + \hat{l}_{bk}[\mathcal{P}_b] \leq l_{rk}[\hat{\mathcal{P}}_r] + l_{bk}[\mathcal{P}_b]. \quad (6)$$

We are going to prove this by distinguishing two cases: 1) $\hat{l}_{rk}[\hat{\mathcal{P}}_r] > l_{bk}[\mathcal{P}_b]$. Since in $\hat{\gamma}$, by definition $\hat{\mathcal{P}}_r$ is a single pocket induced by r , it follows that $\hat{l}_{rk}[\hat{\mathcal{P}}_r] = 1$ and thus $l_{bk}[\mathcal{P}_b] = 0$. Observe that $l_{bk}[\mathcal{P}_b] = 0$ means that no block of color k was present in \mathcal{P}_b in the γ coloring, and this implies that $\hat{l}_{bk}[\mathcal{P}_b] = 0$. Furthermore, since a block of color k appears inside $\hat{\mathcal{P}}_r$ in the coloring $\hat{\gamma}$, a block of color k appeared inside $\hat{\mathcal{P}}_r$ in the coloring γ . Thus it holds that $l_{rk}[\hat{\mathcal{P}}_r] \geq 1$ which together establishes Eq. (6). 2) $\hat{l}_{rk}[\hat{\mathcal{P}}_r] \leq l_{bk}[\mathcal{P}_b]$. We only need to show that $\hat{l}_{bk}[\mathcal{P}_b] \leq l_{rk}[\hat{\mathcal{P}}_r]$. To see this, let P_r^1, \dots, P_r^t be the pockets induced by color r in γ , which are contained in $\hat{\mathcal{P}}_r$ (ordered clockwise). Observe that since in $\hat{\gamma}$ each block in \mathcal{P}_b which was of color r in γ gets colored with color b , there are again exactly t such pockets $\hat{P}_b^1, \dots, \hat{P}_b^t$ induced by color b in $\hat{\gamma}$ which are fully contained in \mathcal{P}_b . Next, observe that for any $1 \leq p \leq t$ it holds that $\hat{P}_b^p \subseteq P_r^p$.

Thus if in $\hat{\gamma}$ a block of color k is contained in a pocket \hat{P}_b^p then this block was contained in pocket P_r^p in γ . This indeed implies that $\hat{l}_{bk}[\mathcal{P}_b] \leq l_{rk}[\hat{\mathcal{P}}_r]$ proving Eq. (6) for this second case.

Using Eq. (2) - (6), we obtain

$$\begin{aligned} \hat{l}_{rk} + \hat{l}_{bk} &\stackrel{(2),(3)}{=} \hat{l}_{rk}[\hat{\mathcal{P}}_r] + \hat{l}_{rk}[\partial P \setminus \hat{\mathcal{P}}_r] + \hat{l}_{bk}[\mathcal{P}_b] + \hat{l}_{bk}[\partial P \setminus \mathcal{P}_b] \\ &\stackrel{(4),(5)}{=} \hat{l}_{rk}[\hat{\mathcal{P}}_r] + l_{rk}[\partial P \setminus \hat{\mathcal{P}}_r] + \hat{l}_{bk}[\mathcal{P}_b] + l_{bk}[\partial P \setminus \mathcal{P}_b] \\ &\stackrel{(6)}{\leq} l_{rk}[\hat{\mathcal{P}}_r] + l_{rk}[\partial P \setminus \hat{\mathcal{P}}_r] + l_{bk}[\mathcal{P}_b] + l_{bk}[\partial P \setminus \mathcal{P}_b] \\ &\stackrel{(2),(3)}{=} l_{rk} + l_{bk}. \end{aligned}$$

Furthermore, since color k was chosen arbitrarily, it holds that

$$\sum_{i \in \{1, \dots, |\mathcal{D}|\} \setminus \{r, b\}} (\hat{l}_{ri} + \hat{l}_{bi}) \leq \sum_{i \in \{1, \dots, |\mathcal{D}|\} \setminus \{r, b\}} (l_{ri} + l_{bi}).$$

Because we colored all blocks of color r in \mathcal{P}_b by color b , it follows that $\hat{l}_{rb} = l_{rb} - 1$ and it thus indeed holds that

$$\sum_{1 \leq i < j \leq |\mathcal{D}|} \hat{l}_{ij} < \sum_{1 \leq i < j \leq |\mathcal{D}|} l_{ij},$$

contradicting the assumption that γ has the smallest crossing number. \square

Lemma 6. *For a crossing-free coloring γ using κ colors, let Π_γ be the set of blocks induced by γ . If $\kappa > 1$ then $|\Pi_\gamma| \leq 2(\kappa - 1)$.*

Proof. We prove the lemma by induction on the number of colors. For $\kappa = 2$, since γ is crossing-free it is easy to see that $|\Pi_\gamma| \leq 2$ and thus the lemma holds. Assuming the lemma holds for $\kappa - 1$ colors, we show it also holds for κ colors. For any color i used by γ , let $\mathcal{B}_i \subseteq \Pi_\gamma$ be the set of blocks of color i . If for all i , $|\mathcal{B}_i| \leq 1$, the lemma trivially holds. Otherwise fix i to be a color with $|\mathcal{B}_i| \geq 2$ and observe that the number of pockets induced by color i is $|\mathcal{B}_i|$.

Let $\mathcal{P}_1, \dots, \mathcal{P}_{|\mathcal{B}_i|}$ be the pockets induced by color i . For each such pocket \mathcal{P}_j we create a new coloring γ_j of ∂P , with

$$\gamma_j(x) = \begin{cases} \gamma(x) & \text{if } x \in \mathcal{P}_j \\ i & \text{otherwise,} \end{cases}$$

as illustrated in Fig. 7(b).

Since γ is crossing-free it is easy to see that for any pocket \mathcal{P}_j , the coloring γ_j is also a crossing-free. Denoting the number of colors of γ_j by κ_j , it holds that

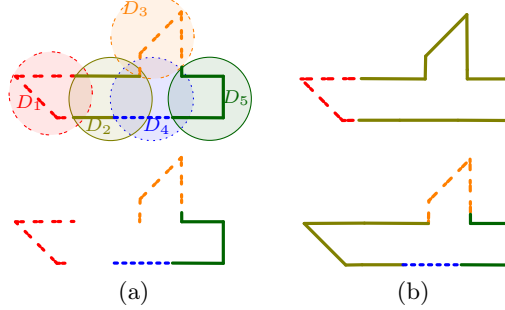


Fig. 7: (a) A disk-coloring example w.r.t. disks D_1, \dots, D_5 ; in the lower part the two pockets induced by color 2 are shown. (b) shows the two polygon colorings in the induction step for color 2 in the proof of Lemma 6.

$1 < \kappa_j < \kappa$, for all $1 \leq j \leq |\mathcal{B}_i|$. Letting Π_{γ_j} be the set of blocks induced by the coloring γ_j , by induction hypothesis $|\Pi_{\gamma_j}| \leq 2(\kappa_j - 1)$.

Observe that each Π_{γ_j} contains exactly one block not in Π_γ . Also, the blocks in \mathcal{B}_i are exactly the blocks not appearing in any of the Π_{γ_j} . Therefore, since the number of pockets induced by color i equals the number of blocks in \mathcal{B}_i , it holds that $|\Pi_\gamma| = \sum_{j=1}^{|\mathcal{B}_i|} |\Pi_{\gamma_j}|$. Thus we obtain

$$|\Pi_\gamma| = \sum_{j=1}^{|\mathcal{B}_i|} |\Pi_{\gamma_j}| \leq 2 \sum_{j=1}^{|\mathcal{B}_i|} (\kappa_j - 1) \quad (7)$$

and because each of the colorings $\gamma_1, \dots, \gamma_{|\mathcal{B}_i|}$ is crossing-free,

$$1 + \sum_{j=1}^{|\mathcal{B}_i|} (\kappa_j - 1) = \kappa, \quad (8)$$

where $(\kappa_j - 1)$ is the number of colors the coloring γ (and also γ_j) uses for the pocket \mathcal{P}_j . The addition of 1 on the left hand side of (3) attributes for color i , which was not counted in any of the pockets. Plugging (3) into (2), the lemma follows. \square

Theorem 1. *The number of disk centers placed by CONTIGUOUSGREEDY is at most $2|OPT| - 1$.*

Proof. If $|OPT| = 1$ then, by its greedy nature, CONTIGUOUSGREEDY also uses only one disk. If $|OPT| > 1$, let γ_{OPT} be a crossing free disk-coloring of ∂P w.r.t. OPT , whose existence is guaranteed by Lemma 5. We let (B_1, B_2, \dots, B_m) be the collection of blocks induced by γ_{OPT} ordered as they appear on ∂P in clockwise order, with B_1 the block containing v_1 . We split B_1 at v_1 into two blocks B_l and B_r , with B_l being the portion of B_1 counterclockwise from v_1 , and $B_r = B_1 \setminus B_l$.

Now observe that by the greedy nature, every disk D computed by CONTIGUOUSGREEDY extends Γ so that $\Gamma \cup D$ fully covers at least one new block in the sequence $(B_r, B_2, \dots, B_m, B_l)$. Therefore, after computing at most $m + 1$ disks, $\Gamma = \partial P$. By Lemma 6, it holds that $m \leq 2(|OPT| - 1)$ and the theorem follows. \square

3.1 Tightness of Analysis

The analysis for the 2-approximation ratio of CONTIGUOUSGREEDY is almost tight, even for convex polygons, as can be seen by a rectangle of length n and height $\epsilon > 0$. It can be covered with $n/(2\sqrt{1 - \epsilon^2/4})$ many geodesic unit disks (by centering them on the median line at height $\epsilon/2$). On the other hand, CONTIGUOUSGREEDY centers disks in steps of 2 on the boundary, thus after finishing one side of the rectangle, each disk introduced a small uncovered hole on the other side. CONTIGUOUSGREEDY covers those holes by placing another $n/2$ disks contiguously on the other side of the polygon, resulting in a total of n disks needed.

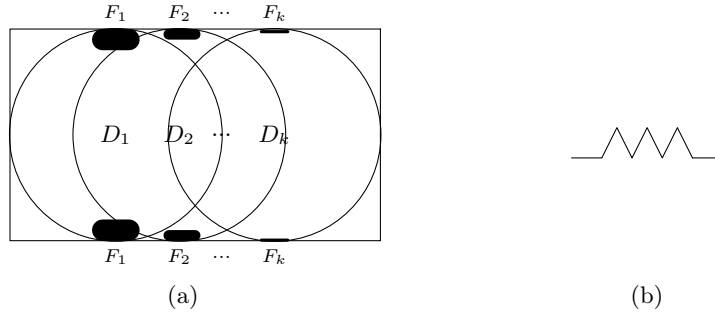


Fig. 8: (a) Illustration of the polygon containing foldings F_1, \dots, F_k on the boundary. A global greedy algorithm starts covering the two F_1 foldings on opposite sides by the disk D_1 , the two F_2 foldings by a disk D_2 and so on, while OPT still only uses a constant number of disks to cover ∂P . (b) Illustration of a folding.

Another natural greedy approach is to cover the largest amount of uncovered boundary at each step. This algorithm results in an approximation ratio of $\Omega(\log n)$, i.e., it is unbounded with respect to $|OPT|$. An example where this greedy rule performs badly is illustrated in Fig. 8(a). The parts of the boundary denoted by F_1, \dots, F_k are dense *foldings* as shown in Fig. 8(b) where the boundary length of F_1 is twice that of F_2 , four times that of F_3 , and so on. The global greedy algorithm first covers the two F_1 sections on opposite sides of the boundary (illustrated by D_1 in Fig. 8(a)), then the two F_2 sections continuing in this way until the two F_k sections are covered, thereby having used k disks to cover the foldings, (plus some constant number of disks to cover the rest of ∂P). Notice that when the height of the polygon is arbitrary close to 2, the number of

foldings can be made arbitrary large, while OPT only uses a constant number of disks to cover ∂P .

It is worth noting that it is crucial that $CONTIGUOUSGREEDY$ *exactly* computes the maximum extension of the contiguous boundary covered by a single geodesic unit disks in each iteration. Only approximately (even with ϵ precision) extending the contiguously covered part by a single geodesic unit disks results in an approximation factor of at least 4 (instead of 2). To see this, we refer to Fig. 9, where c_1 is the endpoint of the ϵ -approximate contiguous greedy extension in the first step and c_1^* is the corresponding exact endpoint (obtained from $CONTIGUOUSGREEDY$). The ϵ -approximate algorithm continues by centering a disks at D_2 which covers the boundary from c_1 up to c_2 . At this point, an exact extension could cover the boundary from c_2 up to c_2^* . However, the approximate algorithm may only cover up to c_3 , by, for example, centering the third disk at D_3 . $CONTIGUOUSGREEDY$ covers up to c_2^* using only two disks. Copying the polygon-section between c_1^* and c_2^* , shows that an ϵ -approximate algorithm performs at least twice as bad as $CONTIGUOUSGREEDY$.

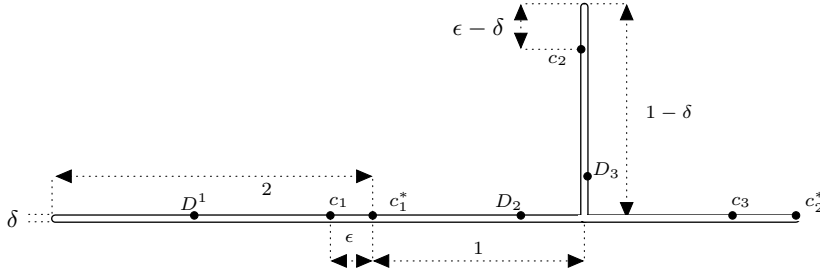


Fig. 9: Illustration of the δ -thin polygon where an ϵ -approximate contiguous extension algorithm results in an approximation ratio larger than 2.

4 Covering Large Perimeters

In this section we show that if the polygon perimeter L is significantly larger than n , i.e., $L \geq n^{1+\delta}$, with $\delta > 0$, a simple linear time algorithm achieves an approximation ratio which goes to one as L/n goes to infinity. For this, we decompose ∂P into long and short portions, based on the length of the corresponding *medial axis*. The medial axis is the set of points in P which have more than one closest point on ∂P . It forms a tree whose edges are either line segments or parabolic arcs and it can be computed in linear time [11]. For a line segment edge, the closest points to the boundary are a subset of two polygon edges; for a parabolic edge, the closest boundary points are a polygon vertex

and a subset of a polygon edge. The idea of the algorithm is to identify long edges of the medial axis (of length at least some constant $c > 2$), and to cover the corresponding polygon boundary section (referred to as *corridors*) almost optimally using only a constant number of disks more than OPT uses to cover the corridor. It is easy to see that each corridor stemming from a parabolic arc can be covered with at most two more disks than OPT uses, by centering disks at distance 2 from each other on the corresponding polygon boundary segment and one disk on the corresponding polygon vertex. Each corridor consisting of a pair of polygon boundary segments can be covered by greedily centering disks on the corresponding medial axis as long as each disk contains corridor portions of length more than two; if the length becomes two or less, greedily center the disks on corridor segments in steps of two. Observe that also in this case, the number of disks needed to cover a corridor is at most two more than OPT uses and their centers can be computed in time linear in their number. This holds since there is at most one point where the covering changes from centering disks on the medial axis to centering disks on ∂P . The rest of the polygon, i.e., the short portions, can be covered greedily by centering $O(n)$ disks on ∂P .

Let \mathcal{D} be the set of all disks placed by the algorithm, $\mathcal{D}_L \subseteq \mathcal{D}$ the disks covering the corridors and $\mathcal{D}_S \subseteq \mathcal{D}$ the $O(n)$ disks covering the short portion of ∂P . Since the number of edges in the medial axis is $O(n)$ (see [11]) and the procedure for covering the long corridors uses at most two more disks than OPT for each corridor, $|\mathcal{D}_L| \leq |OPT| + O(n)$. It therefore holds that $|\mathcal{D}| = |\mathcal{D}_L| + |\mathcal{D}_S| \leq |OPT| + O(n)$. It is easy to see that the disks of OPT which contain a polygon vertex cover at most an $O(n)$ portion of ∂P implying that $|OPT| = \Omega(L)$. Therefore, the approximation ratio can be written as

$$\frac{|\mathcal{D}|}{|OPT|} \leq 1 + \frac{O(n)}{|OPT|} = 1 + \frac{O(n)}{\Omega(L)} = 1 + O\left(\frac{n}{L}\right) = 1 + o(1).$$

5 Acknowledgments

We thank Alon Efrat for his idea of looking at large perimeter polygons. We further thank the anonymous referees for their review of a previous version of this manuscript.

References

1. H. Alt, S. Cabello, P. Giannopoulos, and C. Knauer. On some connection problems in straight-line segment arrangements. In *EuroCG*, 2011.
2. B. Aronov. On the geodesic voronoi diagram of point sites in a simple polygon. *Algorithmica*, 4(1-4):109–140, 1989.
3. B. Aronov, S. Fortune, and G. Wilfong. The furthest-site geodesic voronoi diagram. *Discrete and Computational Geometry*, 9(1):217–255, 1993.
4. L. Barba, P. Bose, M. Korman, J.-L. De Carufel, H.-K. Ahn, and E. Oh. A linear-time algorithm for the geodesic center of a simple polygon. SoCG ’15, 2015. (accepted).
5. S. Bereg and D. G. Kirkpatrick. Approximating barrier resilience in wireless sensor networks. In *ALGOSENSORS*, volume 5804, pages 29–40. Springer, 2009.

6. M. Borgelt, M. Kreveld, and J. Luo. Geodesic disks and clustering in a simple polygon. In T. Tokuyama, editor, *Algorithms and Computation*, volume 4835 of *Lecture Notes in Computer Science*, pages 656–667. Springer Berlin Heidelberg, 2007.
7. P. Brass, W. Moser, and J. Pach. *Research Problems in Discrete Geometry*. Springer, 2005.
8. S. Cabello and P. Giannopoulos. The complexity of separating points in the plane. In *Proceedings of the Twenty-ninth Annual Symposium on Computational Geometry*, SoCG '13, pages 379–386, New York, NY, USA, 2013. ACM.
9. A. Chen, S. Kumar, and T. Lai. Local barrier coverage in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 9(4):491–504, 2010.
10. A. Chen, T. H. Lai, and D. Xuan. Measuring and guaranteeing quality of barrier-coverage in wireless sensor networks. In *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '08, pages 421–430, New York, NY, USA, 2008. ACM.
11. F. Chin, J. Snoeyink, and C. Wang. Finding the medial axis of a simple polygon in linear time. In J. Staples, P. Eades, N. Katoh, and A. Moffat, editors, *Algorithms and Computations*, volume 1004 of *Lecture Notes in Computer Science*, pages 382–391. Springer Berlin Heidelberg, 1995.
12. A. Dumitrescu and C. D. Tóth. Light orthogonal networks with constant geometric dilation. *Journal of Discrete Algorithms*, 7(1):112 – 129, 2009.
13. S. Funke, A. Kesselman, F. Kuhn, Z. Lotker, and M. Segal. Improved approximation algorithms for connected sensor cover. *Wirel. Netw.*, 13(2):153–164, Apr. 2007.
14. M. Gibson, G. Kanade, and K. Varadarajan. On isolating points using disks. In *Proceedings of the 19th European Conference on Algorithms*, ESA'11, pages 61–69, Berlin, Heidelberg, 2011. Springer-Verlag.
15. L. Guibas, J. Hershberger, D. Leven, M. Sharir, and R. Tarjan. Linear-time algorithms for visibility and shortest path problems inside triangulated simple polygons. *Algorithmica*, 2(1-4):209–233, 1987.
16. L. J. Guibas and J. Hershberger. Optimal shortest path queries in a simple polygon. *Journal of Computer and System Sciences*, 39(2):126 – 152, 1989.
17. C.-F. Huang and Y.-C. Tseng. The coverage problem in a wireless sensor network. In *Proceedings of the 2Nd ACM International Conference on Wireless Sensor Networks and Applications*, WSNA '03, pages 115–121, New York, NY, USA, 2003. ACM.
18. R. Kershner. The number of circles covering a set. *American Journal of Mathematics*, pages 665–671, 2010.
19. R.-S. Ko. The complexity of the minimum sensor cover problem with unit-disk sensing regions over a connected monitored region. *IJDSN*, 2012, 2012.
20. S. Kumar, T. H. Lai, and A. Arora. Barrier coverage with wireless sensors. In *Proceedings of the 11th Annual International Conference on Mobile Computing and Networking*, MobiCom '05, pages 284–298, New York, NY, USA, 2005. ACM.
21. B. Liu, O. Dousse, J. Wang, and A. Saipulla. Strong barrier coverage of wireless sensor networks. In *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '08, pages 411–420, New York, NY, USA, 2008. ACM.
22. J. O'Rourke. *Art Gallery Theorems and Algorithms*. Oxford University Press, Inc., New York, NY, USA, 1987.
23. R. Penninger and I. Vigan. Point set isolation using unit disks is np-complete. In *Fall Workshop of Computational Geometry*, 2012.

24. R. Pollack, M. Sharir, and G. Rote. Computing the geodesic center of a simple polygon. *Discrete and Computational Geometry*, 4:611–626, 1989.
25. A. Saipulla, C. Westphal, B. Liu, and J. Wang. Barrier coverage of line-based deployed wireless sensor networks. In *INFOCOM*, pages 127–135. IEEE, 2009.
26. C. Shen, W. Cheng, X. Liao, and S. Peng. Barrier coverage with mobile sensors. In *International Symposium on Parallel Architectures, Algorithms, and Networks, 2008. I-SPAN 2008.*, pages 99–104, May 2008.
27. S. Slijepcevic and M. Potkonjak. Power efficient organization of wireless sensor networks. In *IEEE International Conference on Communications, 2001. ICC 2001.*, volume 2, pages 472–476, 2001.
28. G. T. Toussaint. Computing geodesic properties inside a simple polygon. *Rev. Intell. Artific.*, 1989.
29. I. Vigan. Packing and Covering a Polygon with Geodesic Disks. Proceedings of the 1st Mexican Conference on Discrete Mathematics and Computational Geometry, MCDMCG13, pages 243–252, 2013.
30. B. Wang, H. Xu, W. Liu, and H. Liang. A novel node placement for long belt coverage in wireless networks. *IEEE Transactions on Computers*, 62(12):2341–2353, 2013.